

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-366379

(43)Date of publication of application : 20.12.2002

(51)Int.Cl.

G06F 11/00

(21)Application number : 2001-176108

(71)Applicant : NEC CORP

NEC SYSTEM TECHNOLOGIES LTD

(22)Date of filing : 11.06.2001

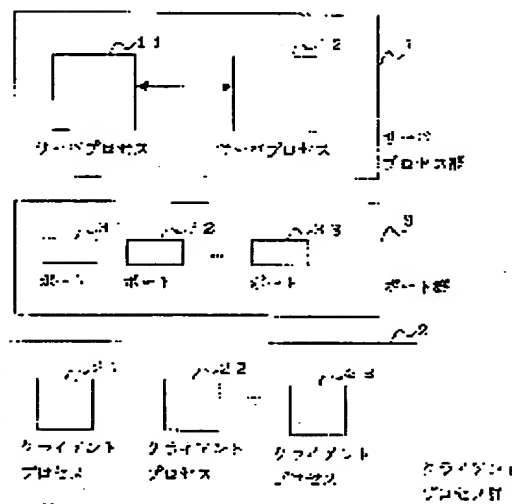
(72)Inventor : ISHIZAWA KATSUYUKI
YOSHIIHARA JUN

(54) SERVICE SYSTEM FOR SERVER PROCESS

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a service system for server process, which doesn't omit services requested from a client process.

SOLUTION: When substitution of an existing server process 11 is requested, a new server process 12 is generated, and the server process 12 describes 'service transfer request' in a shared memory to transmit a signal to the server process 11 when completing preparations for services and waits for a response from the server process 11 after signal transmission. The server process 11 transmits a signal and vanishes by itself after completing services being in the middle of processing. The server process 12 refers to the shared memory when receiving the signal and finds the service transfer response to release the shared memory and starts monitoring all ports 31 to 3n of a port group 3 to provide services.



LEGAL STATUS

[Date of request for examination] 28.05.2002

[Date of sending the examiner's decision of rejection] 16.05.2006

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2002-366379
(P2002-366379A)

(43) 公開日 平成14年12月20日 (2002. 12. 20)

(51) Int.Cl.⁷

識別記号

F I

テマコード (参考)

G 0 6 F 11/00

G 0 6 F 9/06

6 3 0 C 5 B 0 7 6

審査請求 有 請求項の数 6 O L (全 12 頁)

(21) 出願番号 特願2001-176108(P2001-176108)

(22) 出願日 平成13年6月11日 (2001. 6. 11)

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(71) 出願人 390001395

エヌイーシーシステムテクノロジー株式会社

大阪府大阪市中央区城見1丁目4番24号

(72) 発明者 石澤 克之

東京都港区芝五丁目7番1号 日本電気株式会社内

(74) 代理人 100084250

弁理士 丸山 隆夫

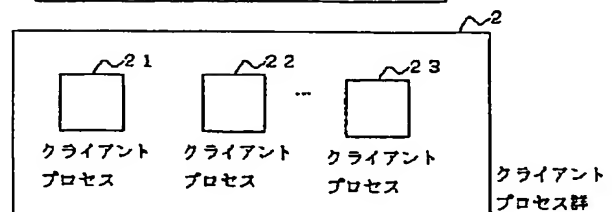
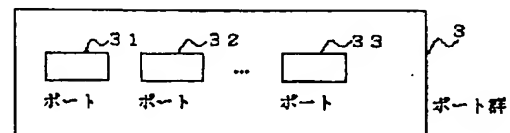
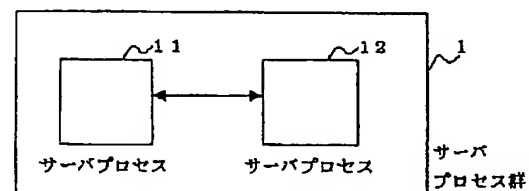
最終頁に続く

(54) 【発明の名称】 サーバプロセスのサービス方式

(57) 【要約】

【課題】 クライアントプロセスから要求されるサービスの取りこぼしの無いサーバプロセスのサービス方式を得る。

【解決手段】 既存のサーバプロセス11に置き換えの要求が発生すると、新しいサーバプロセス12を生成し、サーバプロセス12は、サービスが行えるよう準備ができれば、共有メモリに「サービス移管要求」と記述してサーバプロセス11に対してシグナルを送信し、シグナルを送信した後、サーバプロセス11からの応答を待つ。サーバプロセス11は、シグナルを送信すると、処理途中であるサービスが全て終了した後に、自ら消滅する。サーバプロセス12は、シグナルを受信すると、共有メモリを参照し、サービス移管応答が行われたことを知ると、共有メモリを開放し、ポート群3の全てのポート31～3nの監視を開始してサービスを行う。



【特許請求の範囲】

【請求項1】 クライアントプロセスからの要求毎にセッションの開設／切断を行う構造のサーバプロセスのサービス方式において、

前記サーバプロセスのサーバプログラムをバージョンアップする際に、既存のサーバプロセスから新しいサーバプログラムより生成したサーバプロセスに、前記クライアントプロセスからのサービス要求を待ち受けるポートをオープンし、

該オープンした状態を維持して前記サーバプロセスのサーバプログラムのバージョンアップを実行し、

前記サーバプロセスのサービスに中断を生じさせないことを特徴とするサーバプロセスのサービス方式。

【請求項2】 前記クライアントプロセスからの要求を待ち受けするポートを、前記バージョンアップの実行完了後に既存のサーバプロセスから新しいサーバプロセスに引き継ぐことを特徴とする請求項1記載のサーバプロセスのサービス方式。

【請求項3】 前記既存のサーバプロセスに前記新しいサーバプロセスのサーバプログラムを指定可能としたことを特徴とする請求項1記載のサーバプロセスのサービス方式。

【請求項4】 前記既存のサーバプロセスと前記新しいサーバプロセス間で前記クライアントプロセスからのサービス要求を処理するための同期を取ることを特徴とする請求項1または2に記載のサーバプロセスのサービス方式。

【請求項5】 前記既存のサーバプロセスは、該既存のサーバプロセス自身の行っているサービスを、消滅する前に完結することを特徴とする請求項1から3の何れかに記載のサーバプロセスのサービス方式。

【請求項6】 前記既存のサーバプロセスと新しいサーバプロセスのサーバプロセス同士のやり取りは、シグナル(signal)と共有メモリを利用して行うことを特徴とする請求項2から5の何れかに記載のサーバプロセスのサービス方式。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】 本発明は、サーバプロセスのサービス方式に関し、特に、アクセスへの応答に中断を生じさせないサーバプロセスのサービス方式に関する。

【0002】

【従来の技術】 従来、サーバプロセスのサービス方式は、例えば、サーバプログラムのバージョンアップ時におけるサーバプロセスのサービス方式に関する。

【0003】 従来技術においては、新しいサーバプロセスでのサービスを行うためのバージョンアップ時に、既存のサーバプロセスを一旦終了させる必要がある。例えば、サーバプログラムのバージョンアップ時等に既存のサーバプロセスを終了する際には、クライアントプロセ

スからのサービス要求を待ち受けするポートを、一時的にクローズすることとなる。

【0004】 本発明と技術分野の類似する先願発明例1として、特開2000-315192号公報がある。本先願発明例1では、同一のサービス名のアプリケーションをバージョン毎に管理し、各アプリケーションの管理情報からそれぞれがサービス可能な状態であるか否かを知ることができるようにしている。これにより、クライアントからの処理依頼により指定されたサーバアプリケーションがバージョンアップ中等でサービスができない状況にあっても、既存のサーバアプリケーションに処理を依頼しサービスを停止することなく、継続的なサービスの提供を可能としている。

【0005】

【発明が解決しようとする課題】 しかしながら、上記従来技術では、新しいサーバプロセスでのサービスを行うためには、クライアントプロセスのサービス要求を停止して、サーバプログラムのバージョンアップを行わなければならない。また、既存のサーバプロセスを終了させる際に、既存のサーバプロセスの行っている処理中のサービスを強制的に終了しなければならない。このように、従来技術では、バージョンアップの作業中はサービスが行えない、さらに、場合によっては実行中のサービスを強制終了しなければならない、という問題点がある。

【0006】 なお、先願発明例1では、ポート等が変更された後のアクセスでは、提供された管理情報に基づき新たな手順での処理をしなければならない。

【0007】 本発明は、クライアントプロセスから要求されるサービスの取りこぼしの無いサーバプロセスのサービス方式を提供することを目的とする。

【0008】

【課題を解決するための手段】 かかる目的を達成するため、本発明のサーバプロセスのサービス方式は、クライアントプロセスからの要求毎にセッションの開設／切断を行う構造のサーバプロセスのサービス方式であり、サーバプロセスのサーバプログラムをバージョンアップする際に、既存のサーバプロセスから新しいサーバプログラムより生成したサーバプロセスに、クライアントプロセスからのサービス要求を待ち受けるポートをオープンし、このオープンした状態を維持してサーバプロセスのサーバプログラムのバージョンアップを実行し、サーバプロセスのサービスに中断を生じさせないことを特徴としている。

【0009】 また、上記クライアントプロセスからの要求を待ち受けするポートを、バージョンアップの実行完了後に既存のサーバプロセスから新しいサーバプロセスに引き継ぎ、既存のサーバプロセスに新しいサーバプログラムのサーバプログラムを指定可能とし、既存のサーバプロセスと新しいサーバプロセス間でクライアントプロ

セスからのサービス要求を処理するための同期を取るとよい。

【0010】さらに、上記既存のサーバプロセスは、この既存のサーバプロセス自身の行っているサービスを、消滅する前に完結し、既存のサーバプロセスと新しいサーバプロセスのサーバプロセス同士のやり取りは、シグナル(signal)と共有メモリを利用して行うとよい。

【0011】

【発明の実施の形態】次に、添付図面を参照して本発明によるサーバプロセスのサービス方式の実施の形態を詳細に説明する。図1から図8を参照すると、本発明のサーバプロセスのサービス方式の一実施形態が示されている。

【0012】図1を参照すると、本実施例に適用されるサーバプロセスは、サーバプロセス群1とクライアントプロセス群2、およびサーバプロセスがクライアントプロセスからのサービスを受け取るポート群3から構成される。

【0013】サーバプロセス群1は、サーバプロセス1とサーバプロセス12を含む。サーバプロセス11は、既存のサーバプロセスである。サーバプロセス12は、サーバプロセス11から生成される新しいプログラムのサーバプロセスである。

【0014】全てのサーバプロセス(サーバプロセス11とサーバプロセス12)は、自分とは別のサーバプロセスを生成し、生成したサーバプロセスとサービス無停止置き換えのための情報のやり取りを行うことができる。既存のサーバプロセスが、新しいサーバプロセスを生成する際に利用するプログラムは、プロセス外部から既存のサーバプロセスに指定することができる。

【0015】サーバプロセス同士のやり取りは、シグナル(signal)と共有メモリを利用して行う。このシグナルは、要求が発生したことをプロセスに知らせる。共有メモリには、要求の内容を要求先のプロセスに示すために利用する。プロセスは、シグナルを受け取ったタイミングで共有メモリを参照し、共有メモリに記述されている内容の動作を行う。また、シグナルは、動作を開始するためのトリガとなる。シグナルは、1種類のみを利用する。

【0016】サーバプロセス間の情報のやり取りは、決められたシーケンスで行われる。あるサーバプロセスは、サーバプロセスを生成した時と要求を発行した時に、相手からの応答を待つ必要がある。この待ちの状態には、時間的な上限であるタイムアウト(timeout)時間が設けられている。プロセスが待ちの状態になってこのタイムアウト時間が来ると、置き換えの処理は失敗する。

【0017】共有メモリは、既存のサーバプロセスにより、1つの領域が用意される。置き換えのシーケンスにより、どのサーバプロセスが共有メモリを更新するかが

決められている。このため、異常系における共有メモリ領域に書かれた情報の紛失は、防ぐことができる。

【0018】全てのサーバプロセスは、生成元のプロセスからクライアントプロセスからの要求を待ち受けするポートを、オープンしたまま引き継ぐことができる。クライアントプロセス群2は、クライアントプロセス21~2nを含む。クライアントプロセス21~2nは、サーバプロセスへサービスを要求するプログラムのプロセスである。

【0019】ポート群3は、ポート31~3nを含む。ポート31~3nは、サーバプロセスがクライアントプロセスからのサービスの要求の待ち受けをするためのポートである。

【0020】(動作例)図1、図2を参照して、本実施例の動作について詳細に説明する。既存のサーバプロセス11に置き換えの要求が発生すると(ステップA1)、置き換えの処理が開始される。サーバプロセス11は、新しいサーバプロセスを生成するために起動するプログラムのファイルパスを決定し、共有メモリを確保し、新しいサーバプロセス12を生成する(ステップA2~A4)。

【0021】サーバプロセス11は、サーバプロセス12の生成を行ったら、サーバプロセス12からのサービス移管要求が到着するまで、ポート群3の全ポート31~3nの監視を行い、要求されたサービスを実行する(ステップA5)。

【0022】サーバプロセス12は、サービスが行えるよう準備ができたなら、共有メモリに「サービス移管要求」と記述してサーバプロセス11に対してシグナルを送信する(ステップB1~B3)。サーバプロセス12は、シグナルを送信した後、サーバプロセス11からの応答を待つ(ステップB4)。

【0023】サーバプロセス11は、シグナルを受け取ると、共有メモリを参照し、要求がサービス移管要求であることを知り(ステップA6)、サーバプロセス11はポート群3の全てのポート31~3nの監視をやめる(ステップA7)。サーバプロセス11は、サービス移管要求に応えるため、共有メモリに「サービス移管応答」を記述し、シグナルを送信する(ステップA8~A9)。サーバプロセス11は、シグナルを送信すると、処理途中であるサービスが全て終了した後に、自ら消滅する(ステップA10~A11)。

【0024】サーバプロセス12は、シグナルを受信すると、共有メモリを参照し(ステップB5)、サービス移管応答が行われたことを知ると、共有メモリを開放し(ステップB6)、ポート群3の全てのポート31~3nの監視を開始してサービスを行う(ステップB7)。

【0025】(実施例の効果)第1の効果は、サーバプロセスの置き換えの作業の際にクライアントプロセスのサービス要求を停止しなくてよいことである。その理由

は、ポート情報を引き継ぐことにより、クライアントプロセスからのサービス要求を待ち受けするポートは常にオープンしている（利用可能な）状態であることとなり、クライアントプロセスはサーバプロセスの置き換えを意識することなくサービス要求を行うことができ、クライアントプロセスのサービス要求を停止しなくて済むためである。

【0026】第2の効果は、クライアントプロセスからのサービス要求を取りこぼさないことである。その理由は、サーバプロセスの置き換えの最中にも行われるクライアントプロセスからのサービス要求は、ポートが常にオープンされているため、必ずポートに到達し、既存のサーバプロセスあるいは新しいサーバプロセスのどちらかで対応することとなるためである。

【0027】第3の効果は、置き換え時に既存のサーバプロセスで行われているサービスは完結するまで処理されることである。その理由は、既存のサーバプロセスは、新しいサーバプロセスが通常のサービスを開始した後でも、実行中のサービスを並行して行うことができるからである。

【0028】（第2の実施例）次に、本発明の他の実施例について図面を参照して詳細に説明する。図3を参照すると、サーバプロセス群4がサーバプロセス群1の代わりに存在し、セマフォ群5が新しく存在する点で図1と異なる。サーバプロセス群4は、サーバプロセス41とサーバプロセス42を含む。サーバプロセス41は、既存のサーバプロセスである。サーバプロセス42は、サーバプロセス41から生成される新しいプログラムのサーバプロセスである。

【0029】全てのサーバプロセス（サーバプロセス41とサーバプロセス42）は、自分とは別のサーバプロセスを生成することができる。既存のサーバプロセスが新しいサーバプロセスを生成する際に利用するプログラムは、プロセス外部から既存のサーバプロセスに指定することができる。全てのサーバプロセスは、生成元のプロセスからクライアントプロセスからの要求を待ち受けするポートをオープンしたまま引き継ぐことができる。セマフォ群5は、セマフォ51を含む。

【0030】セマフォ51は、サーバプロセスのポートへのアクセスを排他制御するために利用される。サーバプロセスは、セマフォを得ることでポート群3の全てのポート31～3nにアクセスする権利を得る。セマフォ51を得たサーバプロセスは、ポート群3の全てのポート31～3nの監視を行い、クライアントプロセスからのサービスが到着したら、そのサービスの処理を行う。

【0031】図4のステップA1、A2、A4、A11で示される本実施例におけるサーバプロセス41の動作、および、ステップB1、B8で示される本実施例におけるサーバプロセス42の動作は、それぞれ図1に示されたサーバプロセス11およびサーバプロセス12の

動作と同一のため、説明は省略する。図1では、サーバプロセス11とサーバプロセス12がポート群3の全ポート31～3nの監視を排他的に行うために、プロセス間でシグナルと共有メモリを利用したやり取りを行っていた。本実施例では、ポート群3の全ポート31～3nの監視を排他的に行うために、セマフォを利用する。

【0032】サーバプロセス41は、サーバプロセス42を生成した、取得していたセマフォ51を開放する（図4のステップD1）。サーバプロセス42は、サービスを行う準備が整うと、セマフォ51の取得を試みる。サーバプロセス42は、サーバプロセス41がセマフォ51を開放すると、セマフォ51の取得に成功し（ステップE1）、サービスを行うことができるようになる。

【0033】（第3の実施例）次に、本発明の第3の実施例について図面を参照して詳細に説明する。図5を参照すると、サーバプロセス群6がサーバプロセス群1の代わりに存在している点で図1と異なる。

【0034】サーバプロセス群6は、サーバプロセス群61とサーバプロセス群62を含む。サーバプロセス群61は、管理プロセス611とサービスプロセス612～61nを含む。サーバプロセス群62は、管理プロセス621とサービスプロセス622～62nを含む。

【0035】管理プロセス611および管理プロセス612は、それぞれ、サービスプロセス612～61nおよびサービスプロセス622～62nの管理を行う。管理プロセスは、サービスプロセスを生成／破棄を行いサービスプロセスの数を調整する。管理プロセスは、ポートを監視し、サービスを均等にサービスプロセスに割り振る。例えば、管理プロセス611は、ポート群3の全てのポート31～3nを監視し、クライアントプロセスからサービスの要求があると、サービスプロセス612～61nに均等に要求されたサービスの処理を依頼する。サービスプロセス612～61n、622～62nは、クライアントプロセスから要求されたサービスを行う。

【0036】図6のステップA1～A4、A6～A9、A11、B1～B6で示される本実施例における管理プロセス611および管理プロセス621の動作は、それぞれ、図1に示されたサーバプロセス11およびサーバプロセス12と同一である。このため説明は省略する。

【0037】管理プロセス611は、ステップA4の後、新たなクライアントプロセスからのサービス要求の監視を再開し、サービス要求があればこれに対応する（図6のステップF1）。これを、管理プロセス621からのサービス移管要求が受信されるまで行う。

【0038】ステップA9の後、管理プロセス611は、サービスプロセス612～61nにプロセスの終了を要求する（ステップF2）。各サービスプロセスは、終了の要求を受け取ると、それぞれ行っているサービス

が無くなった後に、自ら消滅する。

【0039】管理プロセス611は、サービスプロセス612～61nの全てのサービスプロセスが消滅するのを待ち（ステップF3）、全てのサービスプロセスが消滅したのを確認したら自らも消滅する。

【0040】管理プロセス621は、ステップB6の後、ポート群3の全ポート31～3nの監視を開始し、サービス要求に対応する（ステップG1）。

【0041】本実施例は、サービスを受け取る機能とサービスを行うを別のプロセスに分けることで、各プロセスの処理を単純にし、サービスの性能およびシステムの保守性を高めている。

【0042】（第4の実施例）次に、本発明の第4の実施例について図面を参照して詳細に説明する。図7を参照すると、サーバプロセス群7がサーバプロセス群4の代わりに存在している点で図3と異なる。サーバプロセス群7は、サーバプロセス群71とサーバプロセス群72を含む。サーバプロセス群71は、管理プロセス711とサービスプロセス712～71nを含む。サーバプロセス群72は、管理プロセス721とサービスプロセス722～72nを含む。

【0043】管理プロセス711および管理プロセス712は、それぞれ、サービスプロセス712～71nおよびサービスプロセス722～72nの管理を行う。管理プロセスは、サービスプロセスを生成／破棄を行い、サービスプロセスの数を調整する。

【0044】サービスプロセス712～71n、722～72nは、セマフォの取得を試み、セマフォを得るとポートの監視を行う。クライアントプロセスからサービス要求が到着すると、セマフォを取得しているサービスプロセスは、要求されたサービスを開始すると同時にセマフォを開放する。サービスプロセスは、行っているサービスが終了すると、再びセマフォの取得を試みる。

【0045】図8のステップA1～A2、A4、A11、F2～F3、B1で示される本実施例における管理プロセス711、721の動作は、図1に示されたサーバプロセス11、12、図6に示された管理プロセス611と同一である。このため説明は省略する。管理プロセス721は、サービスの準備が整うと、サービスプロセスの監視を開始する（図8のステップI1）。

【0046】図1、図3、図5、図7において、新しいサーバプロセスの生成は既存のサーバプロセスから行われているが、サーバプロセスの作り方で、新しいサーバプロセスは既存のサーバプロセス以外から生成することも可能である。

【0047】図1、図5において、既存のサーバプロセスと新しいサーバプロセスの間の情報のやり取りを、シグナルをトリガに行っているが、セマフォをトリガに行うことも可能であり、要求の内容を共有メモリに記述している部分をファイルシステム上のファイルを利用

するようにしても可能である。また、複数の種類のシグナルを利用し、シグナルの種類ごとに意味を持たせてもよい。さらに、既存のサーバプロセスと新しいサーバプロセスの間の情報のやり取りは、TCP やUDP を利用して行うことも可能である。

【0048】図5、図7において、管理プロセスとサービスプロセスは、図1、図3のサーバプロセスの機能をプロセスに分割しているが、図1、図3のサーバプロセスの機能をスレッドに分割してもよい。

【0049】図3、図7において、1つのセマフォで全ポートの排他制御を行っているが、ポートを複数のグループに分け、それらのグループ1つに専用のセマフォを割り当ててポートの排他制御を行ってもよい。

【0050】上記の実施形態によれば、クライアントプロセスからの要求毎にセッションの開設／切断を行う構造のサーバプロセスにおいて、サーバプロセスのサーバプログラムをバージョンアップする際に、既存のサーバプロセスから新しいサーバプログラムより生成したサーバプロセスに、クライアントプロセスからのサービス要求を待ち受けるポートをオープンしたまま引き継ぐ。このことで、サーバプロセスの提供するサービスを停止しないで、サーバプロセスの置き換えを行う。

【0051】既存のサーバプロセスは、外部より、新しいサーバプロセスのサーバプログラムのファイルパスの情報と共にプロセスを置き換える要求を受け取ることができる。既存のサーバプロセスは、プロセスの置き換え要求を受け取ると指定されたサーバプログラムより新しいサーバプロセスを生成する。このとき、既存のサーバプロセスは、自分の利用しているクライアントプロセスからのサービス要求を待ち受けるポートを、クローズすることなくオープンしたまま新しいサーバプロセスに引き継ぐ。

【0052】既存のサーバプロセスと新しいサーバプロセスは、同期を取りながらポートへのアクセスを行う処理を、既存のサーバプロセスから新しいサーバプロセスに移行する。既存のサーバプロセスは、ポートへのアクセスを行う処理を新しいサーバプロセスに移行すると、自分の行っている全てのサービスを最後まで行った後に、自ら消滅する。新しいサーバプロセスは、ポートへのアクセスを行う処理を移行されると、クライアントプロセスからのサービス要求への対応を開始する。

【0053】本発明の特徴は、既存のサーバプロセスと新しいサーバプロセスとの間で、クライアントプロセスからのサービス要求を待ち受けるポートをオープンしたまま引き継ぐこと、既存のサーバプロセスは置き換え時に行っているサービスを完結すること、である。

【0054】上記の実施形態によれば、ポートをオープンしたまま引き継ぐことにより、クライアントプロセスはサーバプロセスの置き換えを意識することなくサービス要求を行うことができ、クライアントプロセスのサー

ビス要求を停止しなくて済む。このことにより、サーバプロセスの置き換えの最中にも行われるクライアントプロセスからのサービス要求においても、ポートが常にオープンされている。このため、必ずポートに到達し既存のサーバプロセス、あるいは新しいサーバプロセスのどちらかで対応する。このように、クライアントプロセスから要求されるサービスを取りこぼすことはない。

【0055】ポートの監視を行わなくなった既存のサーバプロセスは、新しいサーバプロセスが通常のサービスを開始した後も実行中のサービスを並行して行うことができるので、これら実行中のサービスを完結して消滅することができる。このように、サーバプロセスの都合によりサービスが不正に終了することはない。

【0056】尚、上述の実施形態は本発明の好適な実施の一例である。但し、これに限定されるものではなく、本発明の要旨を逸脱しない範囲内において種々変形実施が可能である。

【0057】

【発明の効果】以上の説明より明かなように、本発明のサーバプロセスのサービス方式は、サーバプロセスのサーバプログラムをバージョンアップする際に、既存のサーバプロセスから新しいサーバプログラムより生成したサーバプロセスに、クライアントプロセスからのサービス要求を待ち受けるポートをオープンし、このオープンした状態を維持してサーバプロセスのサーバプログラムのバージョンアップを実行している。よって、サーバプ

ロセスの置き換えの作業の際にクライアントプロセスのサービス要求を停止しなくてよい。

【図面の簡単な説明】

【図1】本発明のサーバプロセスのサービス方式の実施形態を説明するための図である。

【図2】処理手順例を示すフローチャートである。

【図3】第2の実施例を説明するための図である。

【図4】第2の実施例の処理手順例を示すフローチャートである。

【図5】第3の実施例を説明するための図である。

【図6】第3の実施例の処理手順例を示すフローチャートである。

【図7】第4の実施例を説明するための図である。

【図8】第4の実施例の処理手順例を示すフローチャートである。

【符号の説明】

1、4、6、7、61、62、71、72 サーバプロセス群

11、12、41、42 サーバプロセス

2 クライアントプロセス群

21～2n クライアントプロセス

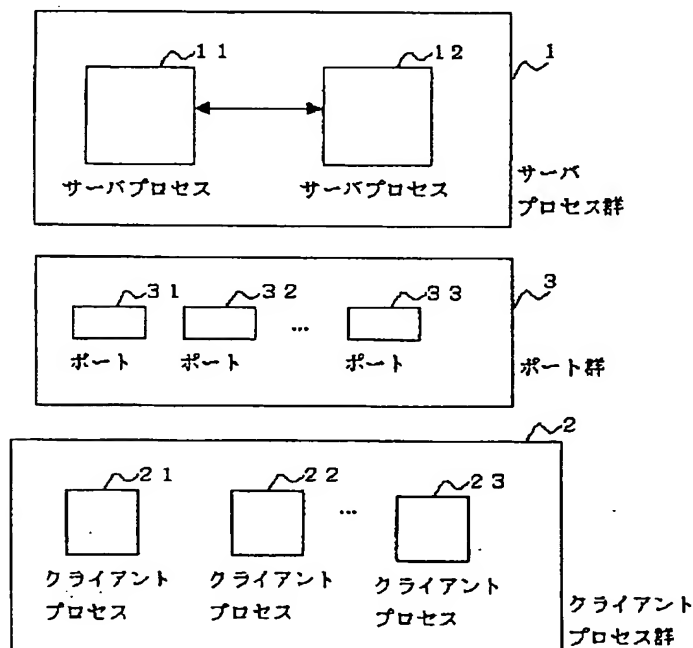
3 ポート群

31～3n ポート

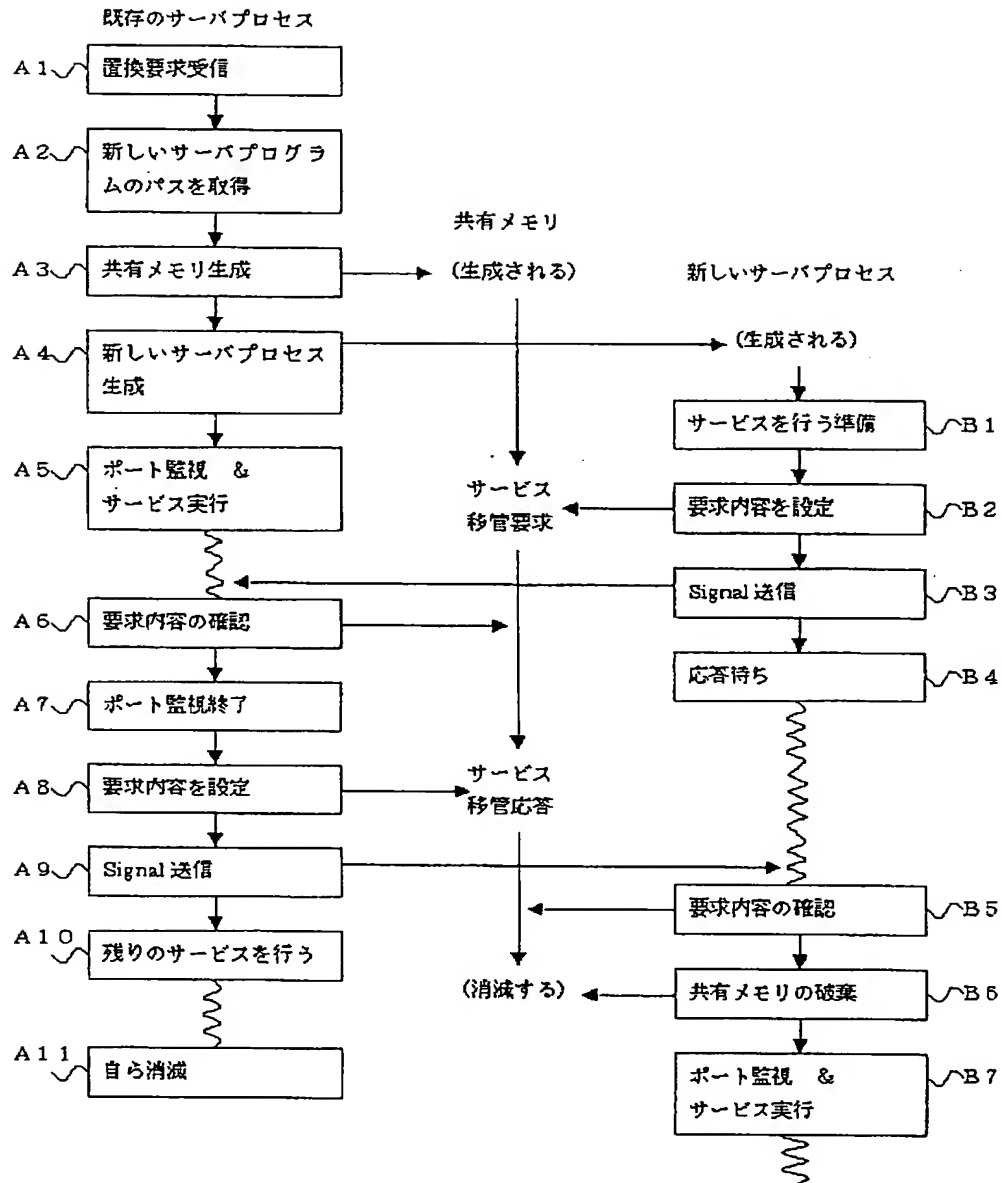
5 セマフォ群

51 セマフォ

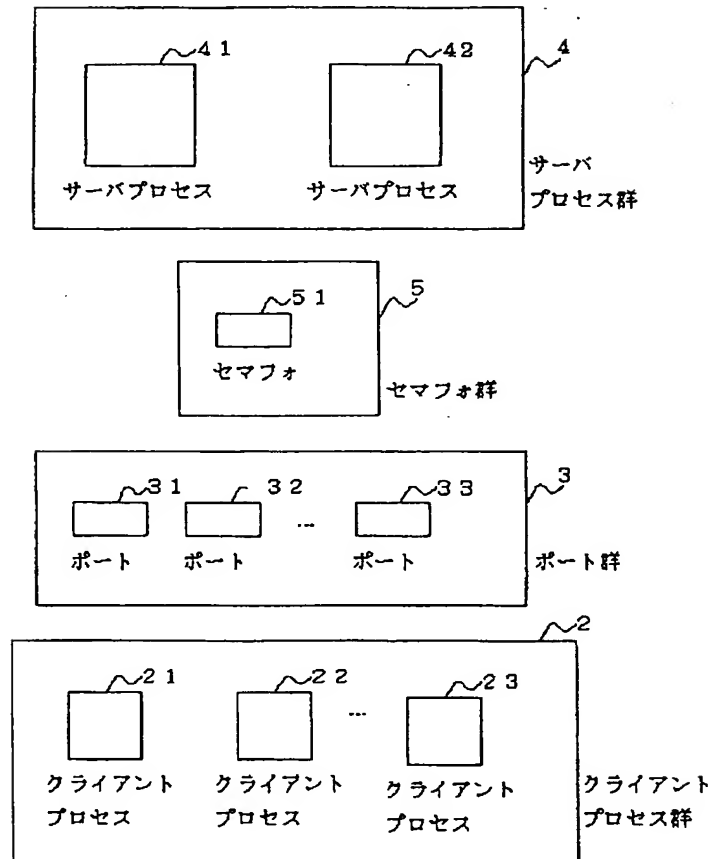
【図1】



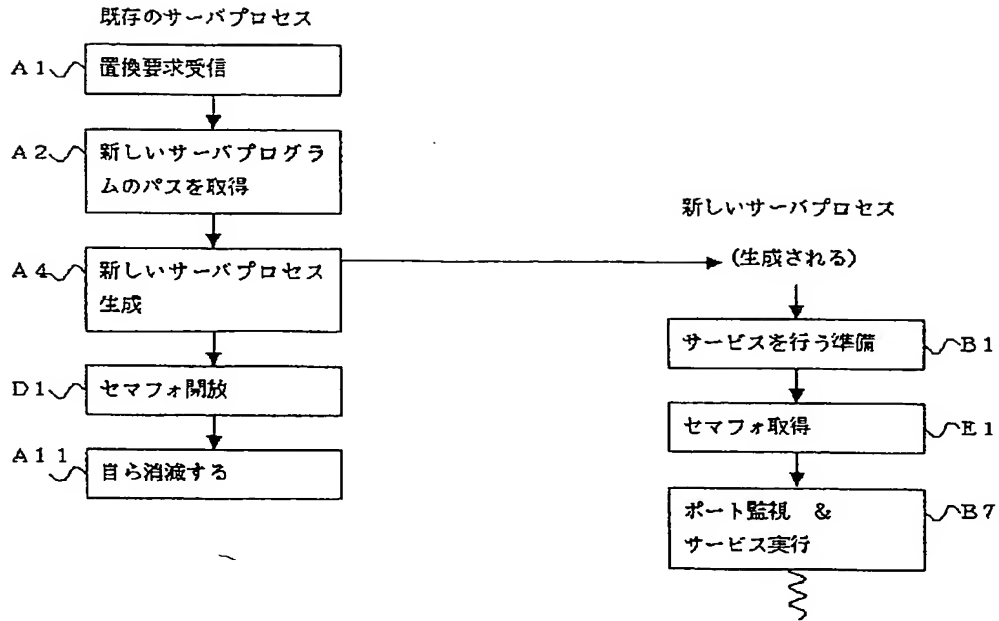
【図 2】



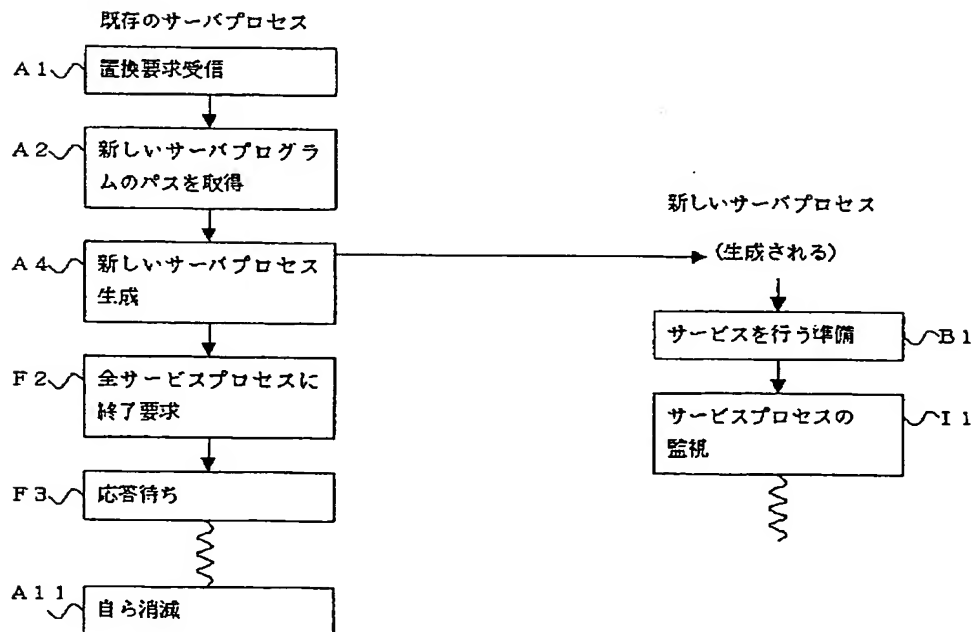
【図3】



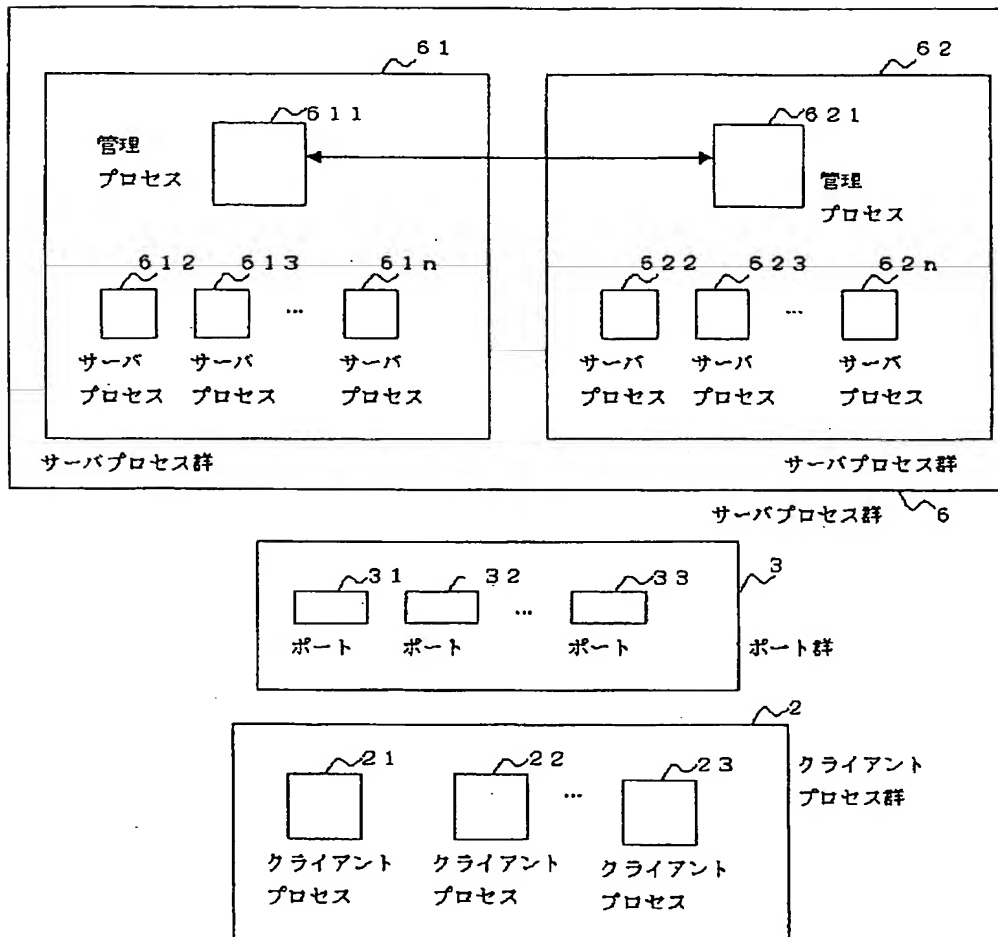
【図 4】



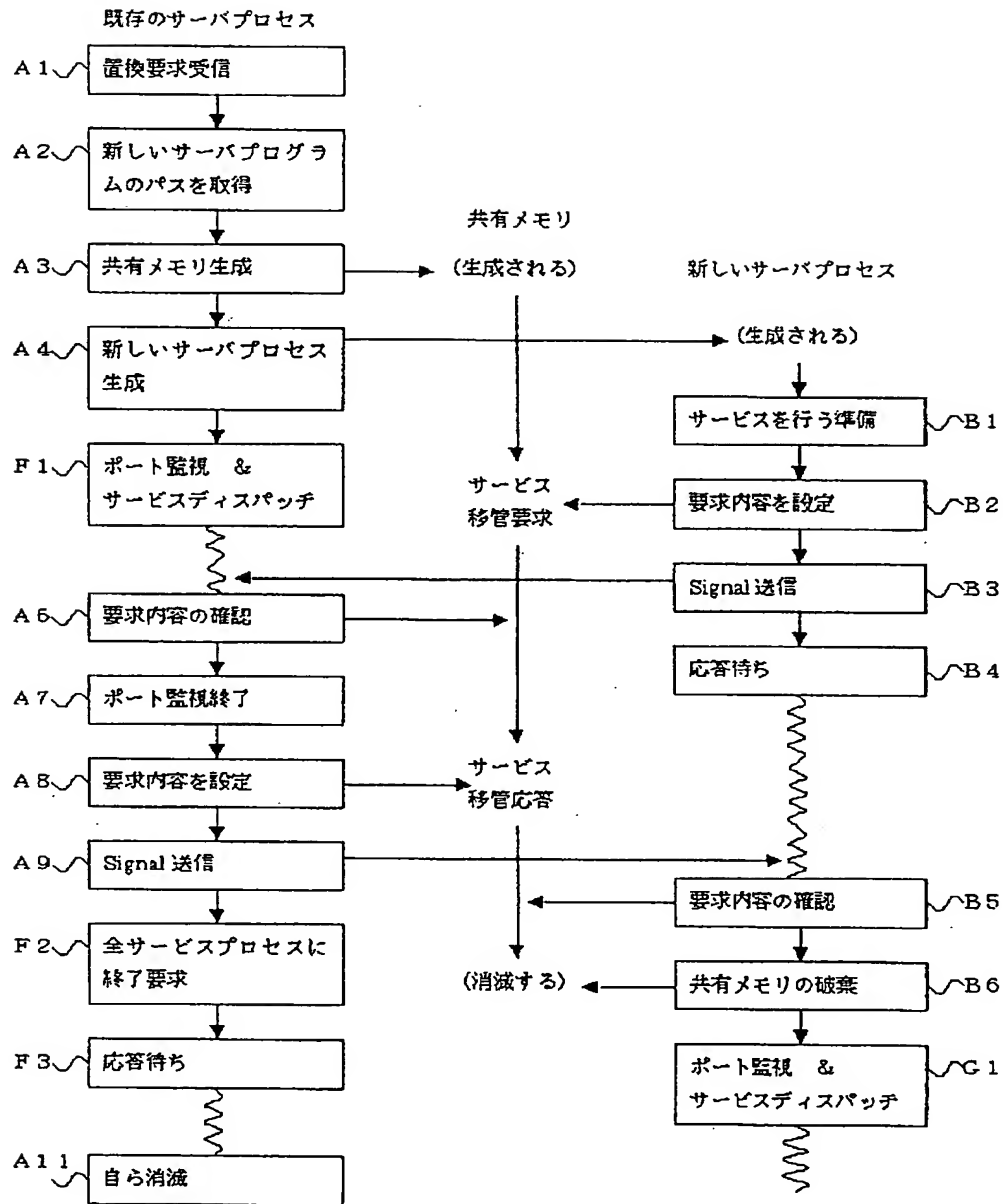
【図 8】



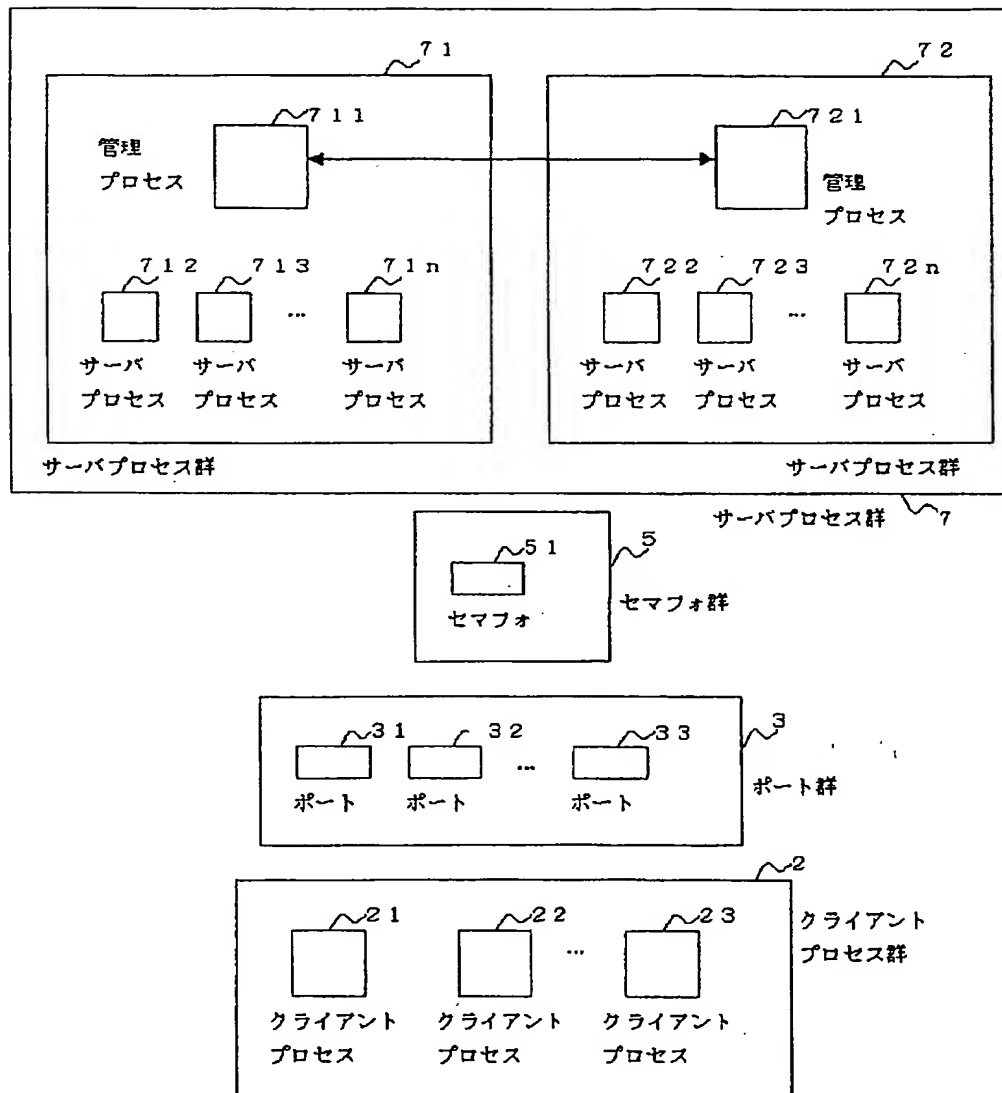
【図5】



【図6】



【図7】



フロントページの続き

(72) 発明者 吉原 潤
 広島県広島市南区稲荷町4番1号 中国日
 本電気ソフトウェア株式会社内

Fターム(参考) 5B076 EA02 EA17